

신경망을 이용한 차량 객체의 그림자 제거

Cast-Shadow Elimination of Vehicle Objects Using Backpropagation Neural Network

정성환*
(Sung-Hwan Jeong)

이준환**
(Jun-Whoan Lee)

요 약

비디오를 이용한 비전기반 감시에서 움직이는 객체의 추적은 GMM (Gaussian Mixture Model)을 사용한 배경영상과 현재영상의 차이법을 이용한다. 문턱치를 통해 생성된 이진영상을 이용하여 객체 추적을 할 경우 객체 정보가 아닌 그림자에 의하여 객체가 병합되는 현상이 나타난다. 본 논문에서는 신경망(Backpropagation Neural Network)을 이용하여 그림자를 제거하는 방법을 제안하였다. 10개의 동영상에서 객체영역과 캐스트그림자(Cast-Shadow)영역의 혼련용 이미지에서 특징 값을 추출하여 신경망을 훈련시켰다. 캐스트그림자를 제거하는 방법은 이진영상의 객체로 추정되는 영역에서 그림자를 분리하는 방법을 기초로 하며 기존의 그림자 제거 알고리즘(SNP, SP, DNMI, DNMI2, CNCC)보다 그림자 제거 성능이 (16.2%, 38.2%, 28.1%, 22.3%, 44.4%)로 높게 나타났다.

Abstract

The moving object tracking in vision based observation using video uses difference method between GMM(Gaussian Mixture Model) based background and present image. In the case of tracking object using binary image made by threshold, the object is merged not by object information but by Cast-Shadow.

This paper proposed the method that eliminates Cast-Shadow using backpropagation Neural Network. The neural network is trained by abstracting feature value form training image of object range in 10-movies and Cast-Shadow range. The method eliminating Cast-Shadow is based on the method distinguishing shadow from binary image, its performance is better(16.2%, 38.2%, 28.1%, 22.3%, 44.4%) than existing Cast-Shadow elimination algorithm(SNP, SP, DNMI, DNMI2, CNCC).

Key Words : Shadow elimination, shadow feature extraction, shadow removal, backpropagation neural network

본 연구는 BK21사업의 연구에 의해 진행됨.

* 주저자 : 전북대학교 전자정보공학부 컴퓨터공학과 박사과정

** 공저자 : 전북대학교 전자정보공학부 교수(교신저자)

† 논문접수일 : 2007년 11월 28일

I. 서 론

교통량 분석을 위한 실시간 영상에서 움직이는 객체를 효율적으로 추적하는 문제가 제기 되어 왔다. 객체 추적을 위한 전처리 작업은 연속된 영상에서 객체가 없는 배경을 추출하여 현재 영상과의 차 영상(image difference)을 구하는 방법을 사용한다. 배경 차이법은 고정된 카메라로부터 얻은 영상에서 변화가 없는 배경 영상을 획득 한 후, 그 배경 영상과 다른 화소들을 움직이는 물체로 간주하여 분할하는 방법이다. 이 방법을 사용하여 객체를 추적할 시에 나타나는 문제점은 배경영상과 현재영상간의 이진영상을 얻는 문제에 있어서 문턱치(threshold)가 중요한 요소로 작용한다. 문턱치에 따라 객체와 배경이 효율적으로 분리 되어 이진영상을 기반으로 한 객체를 추적하는데 있어서 그림자 제거 문제는 높은 변수로 작용한다. 문턱치 값에 따라 영상이 이진화 할 시에 강하지 않은 그림자의 경우에는 차량의 정보는 사라지고 객체 주변의 그림자만 남게 되는 문제점이 있다. 따라서, 교통량 분석을 위한 추적 시스템에서 객체를 정확하게 추적하기 위해서는 객체의 그림자를 제거할 필요가 있다. 그림자 제거를 위한 다양한 방법들이 제안되었다. Tsai 등[1]은 NCC(Normalized Cross Correlation)와 CNCC(Color NCC)방법을 이용하여 그림자를 제거하는 방법을 제안하였는데 이 방법은 단순한 칼라모델과 마스크 주변의 픽셀들과 배경영상간의 상관관계를 얻어내는 방법으로서 상관계수가 -1에서 1의 범위로 나타난다. -1에 가까울수록 배경영상과 현재 영상이 전혀 다른 모습을 나타낸다는 의미이고, 1에 가까울수록 배경과 현재영상간의 상관관계가 높아 이때를 그림자라고 판단하지만, 배경과 비슷한 색상의 객체를 판단할 시에 있어서는 구별 능력이 좋지 않은 결과를 보인다. Grest 등[2]은 CNCC(Color Normalized Cross Correlation)를 이용한 방법을 제안하였는데, 여기서는 입력영상의 HSL 칼라정보를 사용하여 현재영상과 배경영상간의 상관 관계를 구하여 그림자를 제거하였으나 Du-Ming Tsai 등[1]과 마찬가지로 그림자가 배경보다 항상 어둡거나 그림자

와 배경모두에 텍스처 정보가 존재 해야 한다는 문제점이 있다. Gu 등[3]은 PCNN(Pulse-Coupled Neural Network)을 이용하여 객체의 내 외부에 묻어있는 그림자를 제거하였는데 이 방법은 객체 내의 그림자는 지우며 객체를 최대한 보호하고 객체 외부의 그림자를 지우는 방법을 기초로 하지만, 움직이는 객체에 있어서 이 방법을 적용할 시에 추적의 정보인 객체의 정보도 손실될 우려가 있다. 이러한 문제를 해결하기 위해 본 논문에서는 BPNN (Back-Propagation Neural Network)을 이용하여 훈련용 영상을 획득하여 R,G,B정보, 마스크의 그라디언트(gradient)정보 등을 이용하여 그림자를 제거하는 방법을 제안하였다.

본 논문의 제2절에서는 그림자의 정의와 그림자를 제거하기 위한 목적, 본 논문에서 사용한 배경추출방법, CNCC를 이용한 그림자 제거 방법을 설명하며, 제3절에서는 신경망의 입력노드 구성과 신경망의 구조를 설명하며, 제4절에서는 실험 및 결과를, 제5절에서는 결론을 요약하였다.

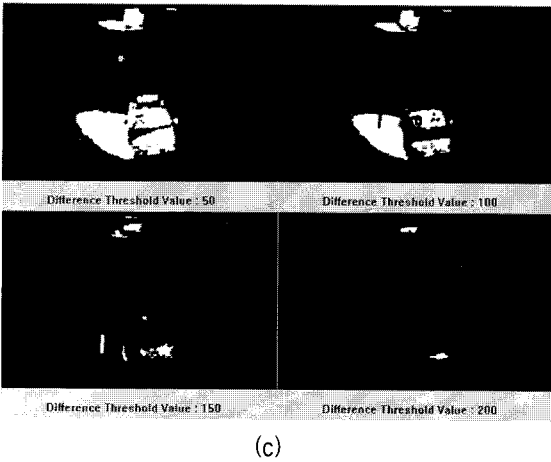
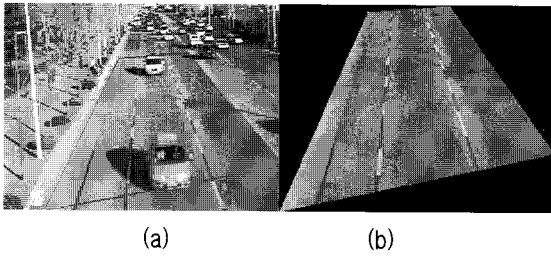
II. 그림자 제거 관련 연구

1. 그림자 종류와 그림자 제거 필요성

객체 추적에 있어서 그림자는 큰 문제가 된다. 그림자를 제거하기 위하여 <그림 1>과 같이 배경이 마지와 현재의 이미지의 차 영상(difference image)의 문턱치 값을 변화시킬 경우 차량까지 지워지는 경우가 발생하여 객체 추적을 위한 이진영상이 사라지는 문제점이 있다.

또한, 문턱치를 낮게 설정하였을 경우 이진 영상을 이용한 객체 추적에 있어서 인접해 있는 두 차량에 있어서 그림자 때문에 두 차량이 병합되는 문제점과 추적에 있어서 그림자와 객체를 모두 객체로 판단하여 추적하는 문제점을 내포하고 있다.

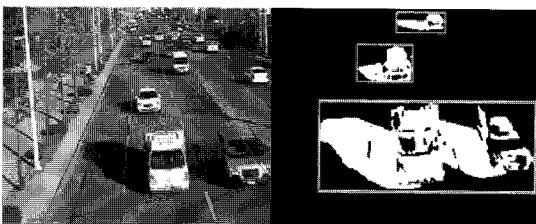
<그림 1>의 (b)는 본 논문에서 사용한 배경생성 방법으로서 많은 논문에서 실험되어 성능이 입증된 Lee[4]가 제안한 GMM(Gaussian Mixture Model)을 사용하였다. GMM은 각 화소를 임의의 개수의 가우시안 분포의 혼합으로 모델링하는 방법이다.



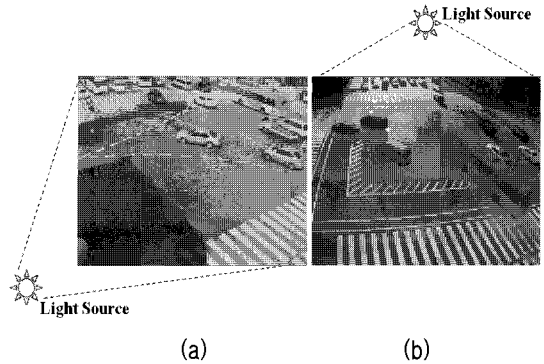
<그림 1> 문턱치에 따른 배경영상과 현재영상의 차이(a) 현재영상 (b) 검지영역내의 배경영상 획득 (c) 문턱치 값을 달리 하였을 때의 이진영상 (T:50,100,150,200)

<Fig. 1> The difference image is based on threshold value (a) The current image (b) The background image (c) The different each threshold value

이 방법에서는 온라인 근사법을 이용하여 모델의 개수를 갱신하는데, 계수의 값을 조절하여 다양한 환경에 적응하는 배경을 추출할 수 있다.



<그림 2> 그림자 때문에 객체가 병합되는 현상
<Fig. 2> The merge of objects and shadows



<그림 3> 그림자의 종류 (a) Still-Shadow(region of blue color) (b) Self-Shadow(region of purple color), Cast-Shadow(region of green color)
<Fig. 3> The kind of shadows (a) Still-Shadow (b) Self-Shadow, Cast-Shadow

관련 연구에 의하면 그림자의 종류는 그림 3과 같이 정지 그림자(Still-Shadow), 자기 그림자(Self-Shadow), 캐스트 그림자(Cast-Shadow)로 구별된다[2][7]. 정지 그림자는 <그림 3>의 (a)와 같이 움직이지 않는 물체에 있어서 광원의 움직임에 의해 변화하는 그림자이며, 자기 그림자는 <그림 3>의 (b)에서 보라색영역으로 광원에 의하여 객체의 반대 방향에 나타나는 그림자 영역으로 캐스트 그림자 영역과는 다르게 객체 부분에 나타나는 그림자이다. Cast-Shadow는 <그림 3>의 (b)에서 연두색영역으로 광원과 객체에 의하여 배경에 나타나는 그림자를 의미한다. 정지 그림자는 배경정보에 포함되어 나타나며, 자기 그림자와 캐스트 그림자는 객체의 이동과 같은 움직임 정보는 지닌다. Cast-Shadow는 객체 추적과 분할 시에 다른 객체에 영향을 끼쳐 캐스트 그림자를 제거해야 한다. <그림 3>의 (b)에서 빨간색 영역은 광원이 객체에 직접적으로 비추는 영역이다.

2. NCC, CNCC(Color NCC)

NCC(Normalized Cross Correlation)와 CNCC(Color NCC)방법은 Tasi 등[1]이 제안한 방법으로 배경이미지와 현재이미지의 상관관계에 있어서 NCC는 gray 이미지, CNCC는 칼라 이미지에서의 상호 상관 관계(cross correlation)를 나타낸다.

본 논문에서는 CNCC방법을 사용하여 본 논문에서 제안한 그림자 제거 방법과 비교 하였다. CNCC 방법은 식(2)와 같이 배경 이미지 $t(i, j)$ 영역과 현재 이미지 $f(x, y)$ 영역간의 칼라 상관관계(δ_c)를 측정하여 그림자를 제거하는 방법이다. 여기서, CNCC는 $T(x, y)$ 와 $F(x, y)$ 의 $m \times n$ 사이즈는 “ $\| \cdot \|$ ” 벡터의 norm이며, δ_c 는 -1에서 1까지를 나타낸다($-1 \leq \delta_c \leq 1$). δ_c 값이 0인 경우 현재 이미지와 배경이미지의 상관도가 애매모호한 경우이다. μ_T 와 μ_F 는 배경이미지(t)와 현재이미지(f)의 $m \times n$ 의 R, G, B 평균이며, $T(x, y)$ 와 $F(x, y)$ 는 다음과 같이 정의된다.

$$\begin{aligned} T(x, y) &= (t_R(x, y), t_G(x, y), t_B(x, y)) \\ F(x, y) &= (f_R(x, y), f_G(x, y), f_B(x, y)) \end{aligned} \quad (1)$$

CNCC방법은 그림자와 배경이미지 간의 텍스처 정보가 존재해야 하며 그림자의 화소가 배경의 화소 보다 어두워야 한다는 것을 가정하고 있다.

$$\delta_c(x, y) = \frac{\sum_{i=-m/2}^{m/2} \sum_{j=-n/2}^{n/2} [F(x+i, y+j) \cdot T(x+i, y+j)^T] - 3 \cdot m \cdot n \cdot \mu_F \cdot \mu_T}{\left\{ \left(\sum_i \sum_j \| F(x+i, y+j) \|^2 - 3 \cdot m \cdot n \cdot \mu_F^2 \right) \cdot \left(\sum_i \sum_j \| T(x+i, y+j) \|^2 - 3 \cdot m \cdot n \cdot \mu_T^2 \right) \right\}^{1/2}}$$

where

$$\begin{aligned} \mu_F &= \frac{1}{3 \cdot m \cdot n} \sum_{i=-m/2}^{m/2} \sum_{j=-n/2}^{n/2} [f_R(x+i, y+j) + f_G(x+i, y+j) + f_B(x+i, y+j)] \\ \mu_T &= \frac{1}{3 \cdot m \cdot n} \sum_{i=-m/2}^{m/2} \sum_{j=-n/2}^{n/2} [t_R(x+i, y+j) + t_G(x+i, y+j) + t_B(x+i, y+j)] \end{aligned}$$

(2)

III. 제안된 방법

본 논문에서 제안한 방법은 비전기반 감시에서 움직이는 캐스트 그림자만을 고려하였고, 알고리즘은 Backpropagation Neural Network을 이용한 그림자 제거 방법이다. 또한, 영상내의 모든 화소를 검사 하지 않고 움직이는 객체와 그림자만을 대상으로 하기 때문에 계산량이 감소하는 이점을 얻을 수 있다.

1. Backpropagation Neural Network(BPNN)

그림자 제거를 위한 관련연구에서 RGB칼라모델을 이용하여 HSV의 칼라모델로 변환하여 그림자를 제거하기 위한 칼라모델을 생성하여 그림자를 제거하는데 사용하였다[5]. 하지만, 이러한 문제는 그림자의 색이 배경과 비슷하거나 배경보다 어두운 경우에만 국한되는 문제점을 보인다.

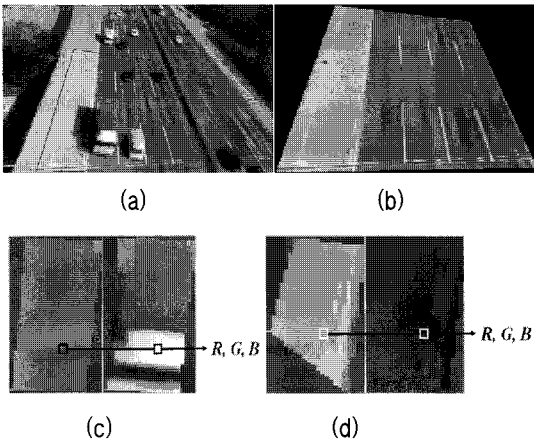
본 논문의 아이디어는 칼라모델을 별도로 설계하는 방법보다는 신경망의 히든노드에서 칼라모델 결정이 네트워크 내에서 이루어 질것이라고 판단하였다. 신경망에서 사용되는 Sigmoid Function이 상관관계 뿐만 아니라 로지스틱함수에 의하여 다양한 통계치를 반영하게 되어 칼라모델이 생성될 것이라고 판단하였다. 또한, 그림자 제거 논문의 기본적인 방법은 그림자와 배경정보가 일치하고 객체와 배경 정보는 일치하지 않는다는 것을 이용하고 있다.

2. 입력노드의 설계

그림자는 해의 위치와 날씨의 변화에 따라 그림자의 농도와 배경의 농도가 변화하게 된다. 따라서,

그림자 제거를 위한 신경망의 훈련데이터는 다양한 상황의 그림자와 객체, 그리고 날씨의 상태를 결정지어줄 배경정보가 필요하다. 신경망의 입력노드는 21개이며 입력노드의 구성은 다음과 같다.

1) 그림자와 객체의 배경의 RGB



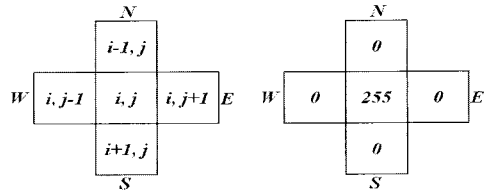
<그림 4> 신경망 훈련을 위한 입력노드 (a) 원본이미지 (b) 배경이미지 (c) 객체와 객체영역의 배경 (d) Cast-Shadow와 Cast-Shadow 영역의 배경

<Fig. 4> The input nodes for neural network training

<그림 4>와 같이 동영상에서 추출한 객체의 이미지, 객체 이미지 영역의 배경, 그림자 이미지, 그림자 이미지 영역의 배경의 R, G, B 화소 값을 이용하였다. 이는 신경망내의 은닉층이 적합한 칼라모델을 결정하고 신경망 훈련을 진행하면서 Binary Sigmoid Function에 의해 2개 이상의 다양한 통계치를 반영하여 적합한 칼라모델을 찾을 수 있다. <그림 4>의 (c), (d)에서 빨간색부분은 객체와 Shadow부분이 아니므로 신경망의 학습능력을 향상시키기 위하여 학습 데이터에 포함하지 않는다.

2) 그림자와 객체에서 배경의 기울기와 평균값

그림자와 객체를 구별 하는데 있어서 그림자에는 배경정보가 어느 정도 존재한다고 가정한다. 객체는 배경을 완전히 다르지만 그림자와 배경은 어느 정도의 유사성을 가지고 있다. 따라서, 배경과 그림자의 정확한 식별을 위하여 <그림 5>와 같이 $x(i, j)$ 에서 N(North), W(West), E(East), S(South)방향의 기울기 정보와 평균값을 이용하였다. 객체는 배경 정보에 비하여 Gradient 값이 크게 나타나며 표면적이 넓은 객체에 대한 구별능력을 향상시키기 위하



<그림 5> NWES방향의 기울기와 최대값

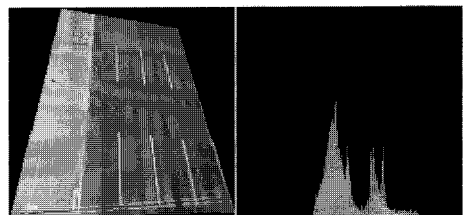
<Fig. 5> The gradient and maximum value of NWES direction

여 평균값을 입력에 같이 사용하였다. R_g, G_g, B_g 는 화소 RGB에 대한 기울기(gradient)이며, $R_{avg}, G_{avg}, B_{avg}$ 는 $x(i, j)$ 에 있어서의 RGB에 대한 평균값을 나타낸다. 이를 식(3)에 나타내었다.

$$\begin{aligned}
 R_g &= |x_r(i, j) - N_r| + |x_r(i, j) - W_r| + |x_r(i, j) - E_r| + |x_r(i, j) - S_r| \\
 G_g &= |x_g(i, j) - N_g| + |x_g(i, j) - W_g| + |x_g(i, j) - E_g| + |x_g(i, j) - S_g| \\
 B_g &= |x_b(i, j) - N_b| + |x_b(i, j) - W_b| + |x_b(i, j) - E_b| + |x_b(i, j) - S_b| \\
 R_{avg} &= (x_r(i, j) + N_r W_r E_r S_r) / 5. \\
 G_{avg} &= (x_g(i, j) + N_g W_g E_g S_g) / 5. \\
 B_{avg} &= (x_b(i, j) + N_b W_b E_b S_b) / 5.
 \end{aligned}
 \tag{3}$$

3) 배경정보

날씨에 따라 그림자의 농도 변화가 다르게 나타나기 때문에 각 상황에서의 그림자의 판별은 매우 어렵다. 해가 질 늦은 오후와 해가 하늘에 떠 있는 낮 시간에는 그림자의 농도가 달라지며 그림자의 크기도 변하게 된다. 본 논문에서는 이 변화를 배경정보를 이용하여 추출하였다. <그림 6>은 검지영역에 있어서의 배경이며, 이 배경의 히스토그램(histogram)을 이용하여 히스토그램의 Dynamic-range, Mean, Variance를 이용하여 신경망의 입력노드로 사용하였다.

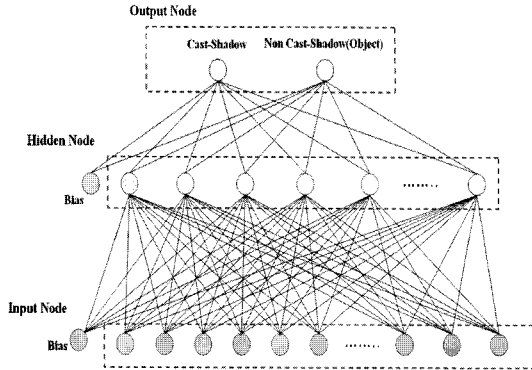


<그림 6> 배경의 히스토그램

<Fig. 6> Histogram of background image

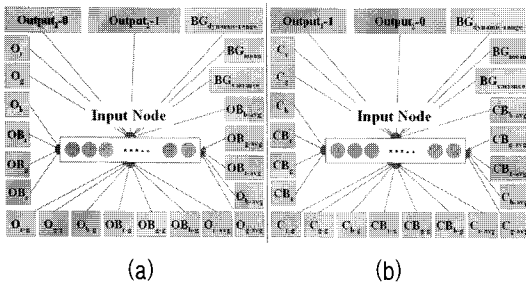
4) 신경망 구조

본 논문에 사용된 신경망은 BPNN(Backpropagation Neural Network)을 이용하였으며, <그림 7>과 같이 입력노드(input node)의 수는 21개, 히든노드(hidden node)의 수는 10개, 출력노드(output node)의 수는 2개로 정의하였다.



<그림 7> 신경망 구조
<Fig. 7> The structure of BPNN

신경망의 입력노드는 <그림 8>과 같이 Object일 때는 원하는 출력이 0, 1이며 Case-Shadow 일 때는 1, 0으로 원하는 출력 값을 지정하였다. 출력 값이 0, 1일 때는 객체를 의미하며 1, 0일 때는 캐스트 그림자를 의미한다.



<그림 8> 신경망의 입력노드
(a) Object (b) Cast-Shadow
<Fig. 8> The input nodes of neural network

<O : Object, C : Cast-Shadow, B : Background>
 O_r, O_g, O_b : Object R, G, B
 OB_r, OB_g, OB_b : Object영역의 배경의 R, G, B

$O_{r-g}, O_{g-g}, O_{b-g}$: Object R, G, B의 기울기(gradient)
 $OB_{r-g}, OB_{g-g}, OB_{b-g}$: Object영역의 배경 R, G, B의 기울기

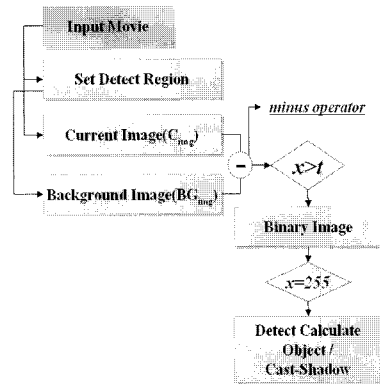
$O_{r-avg}, O_{g-avg}, O_{b-avg}$: Object영역 R, G, B의 평균값
 $OB_{r-avg}, OB_{g-avg}, OB_{b-avg}$: Object영역의 배경 R, G, B의 평균값

$BG_{dynamic-range}, BG_{mean}, BG_{variance}$: 배경(background image) 히스토그램의 dynamic-range, mean, variance

훈련에 사용된 동영상은 오후 2시부터 해가 지지 직전까지의 동영상 10개이며 동영상 한-개의에서 <그림 4>와 같이 무작위로 50개의 Object, Cast-Shadow 이미지를 생성하여 500개의 이미지로 부터 Feature를 추출한 후, 훈련에 사용한 특징 데이터의 수는 총 577,708개 이다.

3. 제안된 방법의 블록도

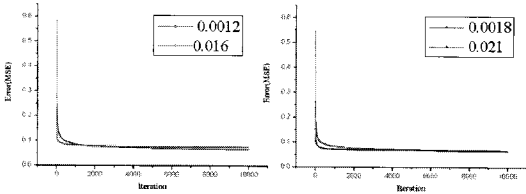
그림자를 제거하는 방법은 <그림 9>와 같다. 동영상에서 검지영역을 지정한 후 지정한 검지영역내의 배경을 실시간으로 얻으며, 얻어진 배경과 현재 이미지의 문턱치를 이용한 이진영상을 생성하고, 생성된 이진영상에서 객체로 나타나는 부분만 그림자 검출 과정을 연산한다. 이와 같이 하면 기존의 이미지 전체를 검사하는 시스템보다 그림자 검출 계산량이 현격히 줄어들게 된다.



<그림 9> 제안된 방법의 블록도
<Fig. 9> The flowchart of suggestion method

IV. 실험 및 결과

훈련 데이터의 MSE(Mean Square Error)값은 <그림 10>과 같다. 학습율과 학습 횟수에 따른 MSE값을 나타낸 것이다. 훈련데이터의 수가 많기 때문에 학습율을 낮게 설정하였으며 학습율이 0.0012일때 MSE값이 0.0636으로 가장 낮은곳에서 수렴하였다.



<그림 10> 학습율의 변화에 따른 MSE 값
<Fig. 10> The MSE value by change of learning rate

훈련된 신경망의 성능 평가를 위하여 본 논문에서 사용된 데이터의 비슷한 시간대의 동영상 3개와 Prati 등[6][8]이 그림자 알고리즘 성능 분석을 위하여 사용한 그림자 제거 실험용 동영상 (highway-raw.avi, <http://cvr.ucsd.edu/aton/shadow>)을 이용하였다.

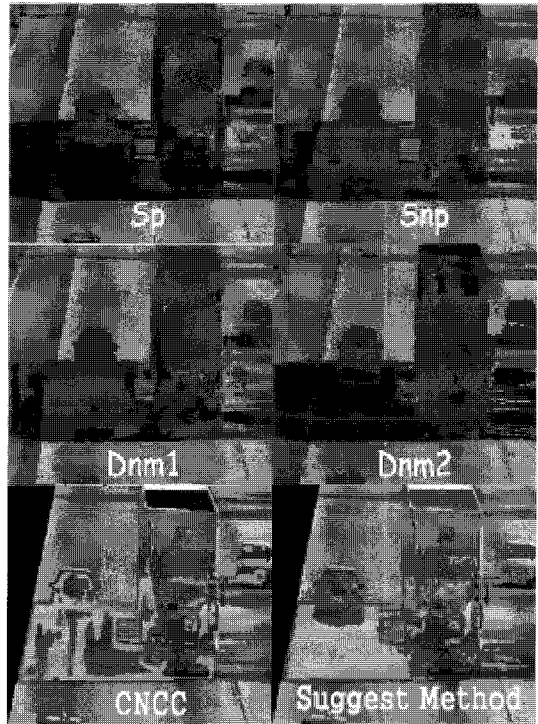
실험에 사용된 데이터는 훈련용 데이터와 마찬가지로 동영상에 대해 배경을 획득하고 동영상 전체에서 50개의 실험용 데이터를 추출 하였다.

Prati 등[6]이 사용한 식(4, 5)과 같은 η 와 ξ 를 산출하여 성능 측정에 사용하였다.

$$\eta = \frac{TP_s}{TP_s + FN_s} \tag{4}$$

$$\xi = \frac{\overline{TP}_F}{TP_F + FN_F} \tag{5}$$

여기서, S는 Shadow, F는 Foreground를 의미하며 TP(true positive), FN(false negative), \overline{TP} 는 foreground 수에서 foreground object가 아닌 shadow 수를 뺀 것을 의미한다. 위의 성능평가 방법을 이용하여 CNCC 방법과, 본 논문에서 제안한 방법, Prati 등[6]이 분석한 방법을 비교하였다.



<그림 11> 그림자 제거 알고리즘별 결과
<Fig. 11> The result image of each other algorithms

<그림 11>은 Patati 논문에서 에서 사용한 highway1-raw.avi 동영상을 테스트한 결과를 나타낸 것이다. SP(Statistical parametric), SNP(Statistical non-parametric), DNM1(Deterministic non-model based), DNM2(Deterministic non-model based), CNCC(Color Normalized Cross Correlation)의 그림자 제거방법을 나타낸 것이다. <그림 11>에서 파란색과 연두색은 캐스트 그림자로 판단한 것이며, 빨간색은 Object로 판단한 결과이다. CNCC은 -1은 배경과 전혀 다른 정보를 가진다는 것을, 1은 배경과 유사한 정보를 가진다는 것을 의미한다. CNCC 방법에서 0보다 큰 값은 그림자, 0보다 작은 값은 객체, 0과 같은 값은 애매모호한 것으로 판정하여 파란색으로 표시한 것이다. 본 논문에서 제안한 방법은 신경망의 output1과 output2값이 0.3이상 차이가 나는 것만 객체와 그림자로 분리하여 그림에 나타나내었다.

<표 1> 6개의 그림자 제거 알고리즘 실험결과, η 그림자 검출 정확도, ξ 그림자 구별 정확도의 비율

<Table 1> Experimental results. Each approach has been tested on the benchmark. Six each experiment the shadow detection accuracy η and the shadow discrimination accuracy ξ in the percentage are reported.

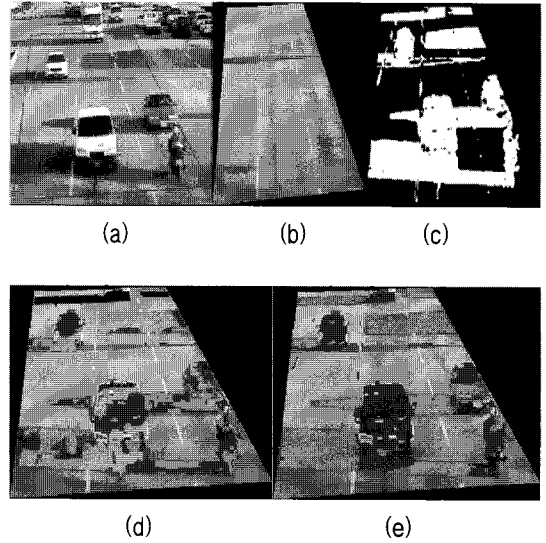
	Highway1- raw.avi	
	η %	ξ %
SNP	81.59%	63.76%
SP	59.59%	84.70%
DNM1	69.72%	76.93%
DNM2	75.49%	62.38%
S-Method	97.83%	94.13%
CNCC	68.30%	49.74%

* S-Method : 본 논문에서 제안한 알고리즘

<표 1>은 Prati 등[6]의 논문에서 평가한 알고리즘 4개와 CNCC방법, 본 논문에 제안한 방법 (Suggest -Method)의 그림자 검출 정확율(η)과 객체와 그림자의 구별 정확율(ξ)을 나타낸 것이다. 본 논문에서 제안한 방법이 97.38%와 94.14%를 나타내어 가장 높은 성능을 보였다.

<그림 12>는 본 논문에서 제안한 방법을 보여주고 있다. 동영상에서 검지영역을 지정한 후 검지영역 내의 배경영상을 획득하고 획득한 문턱치에 따라 배경영상과 현재 영상의 이진 영상을 획득하여 이진영상을 기반으로 하는 그림자 제거 방법을 보여주고 있다. <그림 12>의 (d)는 CNCC방법이며 (e)는 본 논문에서 제안한 방법이다. <그림 12>의 실험에 사용한 동영상은 해가 질 무렵으로 그림자의 농도가 낮아 문턱치를 30으로 하여 차 영상을 생성하여 이용하였다.

<그림 13>은 실험영상 3개를 이용하여 CNCC와 본 논문에서 제안한 방법을 이용하여 그림자와 객체를 분리 한 그림이다. 편의를 위하여 <그림 13>의 위쪽 그림 중 왼쪽을 movie 1, 가운데를 movie 2, 오른쪽을 movie 3으로 명칭 하였다. 이 세 개의 동영상에서의 그림자 제거 성능을 <표 3>에 나타내었다. 본 논문에서 제안한 그림자 제거 방법을 이용하였을 때

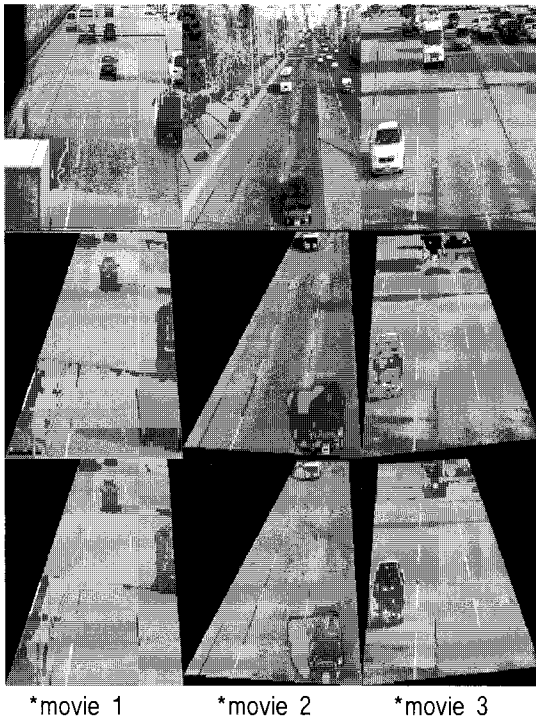


<그림 12> CNCC와 제안 방법의 그림자 제거 알고리즘 결과 (a) 원본 이미지 (b) 배경 (c) 이진영상 (d) CNCC방법 (e) 제안 방법

<Fig. 12> The shadow elimination algorithm results of CNCC and suggestion method (a) original image (b) background image (c) binary image (d) CNCC method (e) suggestion method

의 η, ξ 성능 모두 90%이상의 그림자 구별 성능이 측정되었다. 또한, 픽셀 9267개를 그림자와 객체로 구별하는데 걸리는 시간은 CNCC 방법은 7.43초가 소요되는 반면 본 논문에서 제안한 방법은 0.27초로 초당 5~10프레임의 처리가 가능하다는 결론을 얻을 수 있다. <그림 14>는 교차로에서 얻은 영상으로 CNCC방법과 본 논문에서 제안한 방법을 이용하여 그림자를 제거한 실험 결과이다. <그림 14> (c)는 CNCC를 이용한 그림자 제거 방법으로 파란색 영역은 객체도 그림자도 아닌 애매모호한 결과를 나타내고 있는 것으로 본 논문에서 제안한 방법 (그림 14 (d))보다 좋지 않은 결과를 나타내고 있다.

본 논문에서 사용한 동영상은 크기 320×240, 30frame/s를 이용하였으며, 펜티엄 코어 2 듀오 6420 프로세서, 2기가 램, 윈도우 XP환경에서 Visual C++ 6.0과 DirectShow SDK를 사용하였다.



<그림 13> CNCC와 제안 방법의 그림자 제거 알고리즘 결과, 위:원본 이미지, 중간:CNCC방법, 아래:제안 방법

<Fig. 13> The shadow elimination algorithm results of CNCC and suggestion method, Top:original image, Middle:CNCC method, Bottom : Suggestion method

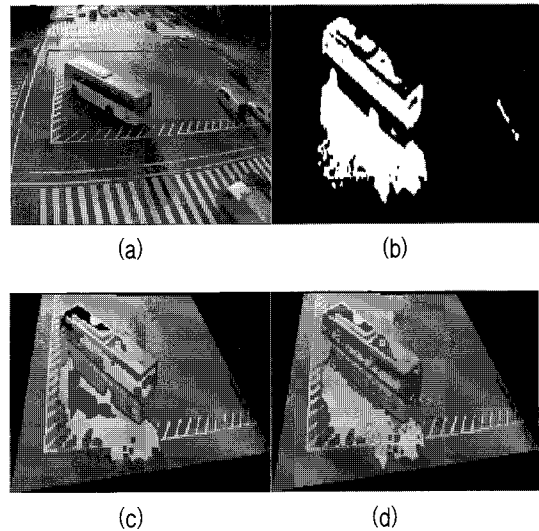
<표 2> 3개의 실험 동영상의 실험 결과

<Table 2> The tested result of 3 test the movie.

	movie 1		movie 2		movie 3	
	$\eta\%$	$\xi\%$	$\eta\%$	$\xi\%$	$\eta\%$	$\xi\%$
S-Method	94.70%	93.23%	99.57%	91.89%	98.58%	92.16%
CNCC	17.49%	35.59%	24.38%	54.01%	50.74%	48.86%

V. 결 론

본 논문에서는 비전기반 감시에서 움직이는 객체의 효율적인 객체 분할을 위하여 신경망을 이용한 캐스트 그림자 제거방법을 제안하였다. 영상 전체가 아닌 검지영역의 배경영상을 획득하고 획득한 배경영상과 현재영상의 문턱치를 이용한 이진 영상



<그림 14> CNCC와 제안 방법의 그림자 제거 알고리즘 결과 (a) 원본 이미지 (b) 이진영상 (c) CNCC방법 (d) 제안방법

<Fig. 14> The shadow elimination algorithm results of CNCC and suggestion method (a) original image (b) binary image (c) The result of CNCC method (d) The result of suggestion method

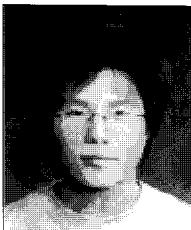
을 생성하여 객체라고 추정되는 그림자와 객체부분을 효율적으로 분리하였다. 그림자 제거에 있어서 다른 그림자 제거 방법보다 높은 성능을 발휘 하였으며 그림자 제거에 소요되는 시간 또한 단축시키는 결과를 얻었다. 이는 비전기반 교통감시에서 사용되는 이진영상을 이용한 객체추적에서 효율적으로 사용될 수 있을 것이다.

본 논문에서 제안한 방법이 보완해야 할 부분은 신경망의 훈련데이터용 영상을 다양하게 하여야 하며 신경망 훈련 시 MSE값을 계속적으로 줄이게 되면 오히려 캐스트 그림자(cast-shadow) 만을 제거하지 못하고 객체나 자기 그림자(self-shadow)까지 제거되는 문제점이 있다. 신경망 훈련 시 데이터의 수에 따른 MSE값을 적절히 조정하여 수렴시켜야 하는 단점을 해결해야 할 것이다.

참고문헌

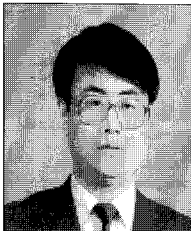
- [1] D. M. Tsai, C. T. Lin, and J. F. Chen, "The evaluation of normalized cross correlations for defect detection," *Pattern Recognition Lett.*, vol. 24, no. 15, pp. 2525-2535, Nov. 2003.
- [2] D. Grest, J. M. Frahm, and R. Koch, "A color similarity measure for robust shadow removal in real-time," *Proc. Vision, Modeling, and Visualization*, pp. 253-260, Nov. 2003.
- [3] X. Gu, D. Yu, and L. Zhang, "Image shadow removal using pulse coupled neural network," *IEEE Trans. Neural Networks*, vol. 16, no. 3, pp. 692-698, May 2005.
- [4] D. S. Lee, "Effective gaussian mixture learning for video background subtraction," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 27, no. 5, pp. 827-832, May 2005.
- [5] R. Cucchiara, C. Grana, M. Piccardi, A. Prati, and S. Sirotti, "Improving shadow suppression in moving object detecting with HSV color information," *Proc. IEEE Intelligent Transportation System Conf.*, pp. 334-339, Aug. 2001.
- [6] A. Prati, I. Mikic, M. M. Trivedi, and R. Cucchiara, "Detecting moving shadow: algorithm and evaluation," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 25, no. 7, pp. 918-923, July 2003.
- [7] C. Jiang and M. O. Ward, "Shadow identification," *Proc. IEEE Int. Conf. Computer Vision and Pattern Recognition*, pp. 606-612, June 1992.
- [8] A. Prati, I. Mikic, C. Grana, and M. M. Trivedi, "Shadow detecting algorithm for traffic flow analysis: a Comparative study," *Proc. IEEE Intelligent Transportation System Conf.*, pp. 340-345, Aug. 2001.

저자소개



정 성 환 (Jeong, SungHwan)

2007년 3월 ~ 현재 : 전북대학교 박사과정 재학(컴퓨터공학과)
 2000년 3월 ~ 2004년 2월 : 전주대학교 컴퓨터공학, 정보통신공학(공학사)
 2004년 3월 ~ 2006년 2월 : 전북대학교 의용생체공학과(공학석사)



이 준 환 (Lee, JoonHwan)

1999년 ~ 현재 : 전북대학교 전자공학과 교수
 1980년 : 한양대학교 전자공학과(공학사)
 1982년 : 한국과학기술원 전자공학과(공학석사)
 1982년 ~ 1985년 : 전북대학교 전자공학과 조교
 1985년 ~ 1987년 : 전북대학교 전자공학과 전임강사
 1990년 : 미주리대학 전산학과(공학박사)